

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended) A system for organizing and accessing a database, the system comprising:
 - a primary B+tree index;
 - a secondary B+tree index;
 - a plurality of mapping table row identifiers stored in the secondary B+tree index, where each mapping row identifier is an entry in a row of the secondary B+tree index and uniquely identifies a row within a mapping table; and
 - a first plurality of database addresses for leaf blocks of the primary B+tree index stored in the secondary B+tree index; and
 - a corresponding number of database addresses for leaf blocks of the primary B+tree index stored in a row of the mapping table, wherein data stored within the primary B+tree is accessed and organized using the secondary B+tree or mapping table corresponding to mapping table rows.
2. (original) The system according to claim 1, wherein 4 bytes of the database addresses are stored in the secondary B+tree index.
3. (currently amended) The system according to claim 1, wherein the database addresses

correspond to a guess-database addresses address, where each guess-database address is a guess as to what address block of the primary B+tree a row may be found.

4. (currently amended) The system according to claim 1, further comprising:
a guess-database address quality statistic for the secondary index and a guess-database address quality statistic for the mapping table, ~~both statistics being operable to assess guess-database address quality, where each of the guess-database address quality statistic represents a ratio of how often the guesses as to where rows may be found in an address block of the primary B+tree are accurate.~~

5. (currently amended) A method for managing a database system, the method comprising:

creating a secondary index for a B+tree structure, wherein the secondary index structure comprises a plurality of rows each comprising an index key value, a mapping table rowid value that uniquely identifies a row within a mapping table, and a guess-database address value that represents an address block of a primary B+tree where a row may be found; and
retrieving data stored in the database system using the secondary index for a B+tree.

6. (original) The method according to claim 5, further comprising:
inserting a row of the secondary index structure, wherein inserting the row comprises inserting a row comprising an index key value, a mapping table rowid value and a guess-database address value.

7. (original) The method according to claim 5, further comprising:

deleting a row of the secondary index, wherein deleting the row comprises locating a row comprising an index key value and a mapping table row identifier and deleting the row.

8. (original) The method according to claim 5, further comprising:

updating the secondary index, wherein updating the secondary index comprises locating a row of the secondary index comprising an old index key and a mapping table row identifier, deleting the row and inserting in the row a new index key value, a mapping table row identifier and a guess database address.

9. (original) The method according to claim 5, further comprising carrying out a query utilizing the secondary index, wherein carrying out the query comprises:

utilizing a guess-database address stored as part of a secondary index row to find a row in the primary B+tree structure;

obtaining a target database block from the row in the primary B+tree structure;

searching the target database block for a row that contains a mapping table row identifier that is the same as a mapping table row identifier stored in the secondary index row; and

if the row in the database block matches the target database block, then the correct row in the database has been located and the query is completed.

10. (original) The method according to claim 9, wherein if the row in the database block does not match the target database block, carrying out the query further comprises:

accessing the mapping table row stored in the secondary index row;

utilizing a guess-database address stored in the mapping table row to access a target block of the database;

searching the target block for a primary key that matches a primary key stored in the mapping table row; and

if the primary key is found, then the query is completed.

11. (original) The method according to claim 10, wherein if the primary key is not located carrying out the query further comprises:

traversing the primary B+tree structure utilizing the primary key value from the mapping table row to identify the database address to complete the query.

12. (currently amended) The method according to claim 11, further comprising:
maintaining a guess-database address quality statistic for the secondary index;
maintaining a guess-database address quality statistic for the mapping table;
utilizing the statistics to assess guess-database address quality; and
carrying out the query based upon guess-database quality in the secondary index and mapping table, where each of the quess-database address quality statistic represents a ratio of how often the guesses as to where rows may be found in an address block of the primary B+tree are accurate.

13. (original) The method according to claim 12, further comprising:

estimating guess-database address quality;

estimating the cost of the query based upon the estimated guess-database address quality;

carrying out the query starting with an index structure with the lowest estimated cost.

14. (original) The method according to claim 5, wherein only 4 bytes of the guess-database address value are stored in the secondary index row.

15. (currently amended) A computer program product for performing a process of managing a database system, comprising:

a computer readable medium; and

computer program instructions, recorded on the computer readable medium, executable by a processor, for performing the steps of:

creating a secondary index for a B+tree structure, wherein the secondary index structure comprises a plurality of rows each comprising an index key value, a mapping table rowid value that uniquely identifies a row within a mapping table and a guess-database address value that represents an address block of a primary B+tree where a row may be found; and

retrieving data stored in a database system using the secondary index for a B+tree.

16. (currently amended) A system for performing a database management process, comprising

a processor operable to execute computer program instructions; and

a memory operable to store computer program instructions executable by the processor, for performing the steps of:

creating a secondary index for a B+tree structure, wherein the secondary index structure

10/790,146

docket no. 19111.0143

comprises a plurality of rows each comprising an index key value, a mapping table rowid value

that uniquely identifies a row within a mapping table and a guess-database address value that

represents an address block of a primary B+tree where a row may be found; and

retrieving data stored in a database system using the secondary index for a B+tree.